

REMARKS

In response to the above-identified Office Action, Applicants seek reconsideration of the claims in light of the following arguments.

I. Amendment to Figure 2

In the previous response dated February 2, 2005, Applicants provided a replacement drawing sheet for Figure 2. However, the Examiner in the above-identified Office Action did not acknowledge the amendment. Applicants respectfully request entry of the drawing amendment.

II. Response to Final Rejection

Applicants will respond to each of the Examiner's arguments in turn.

The Examiner asserted in the Final Office Action (p. 11) that:

Knutson teaches deploying of beans to LAN processors (see Fig. 1A) and further (Fig. 1B) mentions about filtering rules for allowing beans to get identified and verified during their instantiation for a particular business transaction. It was a well-accepted concept that a enterprise or a business subsystems having a LAN always set up rules to control communication of outside data in and out of the enterprise private local network. The inherency of rules controlling what data should be permitted to the processors inside a private enterprise LAN was a fact at the time the invention was made.

Applicants respectfully submit that the Examiner is confusing deployment of EJBs with their runtime behavior. According to Knutson, an EJB “container 114 acts as a filter and provides rules concerning transactions, state, security, etc., on all operations.” Knutson, col. 3, lines 62-64. Thus, the “rules” the Examiner speaks of have to do with governing EJB operations—not deployment. Whether such rules are inherent or not is irrelevant.

The Examiner asserted in the Final Office Action (p. 11) that:

The Applicant has to prove why the presence of an enterprise LAN like that of Knutson otherwise means that regulating or filtering rules (via a portal service or filtering proxy or firewall) are not necessarily present at the time the invention

was made, i.e. can enterprise local network remain local/proprietary without a filtering portal service to enforce internal security/integrity?

Here the Examiner seems to be equating the EJB runtime container rules discussed above with logic in network routers, proxies and firewalls. Obviously, most computer networks include hardware and/or software to perform message routing and filtering. But Applicants fail to see how this is relevant to selecting processing devices for deployment of applications.

The Examiner asserted in the Final Office Action (p. 12) that:

The rejection has pointed to the filtering rules parts which a EJB interface utilizes for enabling creation of session beans for a particular business transaction in a client/server paradigm; and in conjunction with the LAN concept as mentioned above, it is clear that rules that govern how data can be accessed or transmitted to and fro across the gate of a LAN for addressing the need of a particular business transaction reads on the fact that only a portion that satisfies those rules have the ability/permission to access the distributed beans or application in the above enterprise-related client/server paradigm according to the specific transaction and security instantiation thereof. In other words, the LAN and the security aspect of beans in the client/server paradigm thus combined has met the limitation at issue, mostly based on lack of specificity in the claim; and on grounds that inherent teaching can be part of material for anticipating a subject matter.

As a preliminary matter, the Examiner fails to cite a single piece of art or provide any evidence revealing how EJB runtime container rules can be combined with network routers/proxies/firewalls. Second, even if such a combination were possible, the proposition itself is irrelevant since this “inherent” combination (per the Examiner) does not address selecting processing devices for deployment of applications.

The Examiner asserted in the Final Office Action (p. 13) that:

Applicant has submitted that Seidman only deploys to a directory service and fails to teach or suggest ‘to a selected portion ... processing devices’ (Appl. Rmrks, pg. 10, middle). The recited ‘application directory’ does not enforce the interpretation that it mean ‘selected portion’ as claimed; because broadly speaking it entails a file system or a disc arrangement of code or a array thereof in a computer. The rejection has shown that this directory limitation is if not anticipated by Knutson

then would have been obvious in view of Seidman. The Seidman in combination with Knutson has been put forth to address the obviousness of the limitation as to storing of undeployed application in a structure like a directory.

The Examiner apparently asserts in the above text that a “processing device” in Applicants’ claims is equivalent to an “application directory” in Seidman. However, the Examiner cites no support in Seidman that an application directory is capable of executing an application (see Applicants’ independent claims 1, 37 and 40). Therefore, the comparison is unfounded.

The Examiner asserted in the Final Office Action (p. 13) that:

Seidman was never meant to address ‘selected portion’ as alleged by the above arguments; and the argument has to rebut the obviousness type of rejection set forth via such combination. In response to applicant’s arguments against the references individually, one cannot show nonobviousness by attacking references individually where the rejections are based on combinations of references. See *In re Kelle*, 642 F.2d 413, 208 USPQ 871 (CCPA 1981); *In re Merck & Co.*, 800 F.2d 1091, 231 USPQ 375 (Fed. Cir. 1986).

The Examiner is misguided in the application of the law on this point. Applicants have shown that neither reference teaches or suggests the same limitation. Therefore, it follows that their combination would not teach or suggest it.

The Examiner asserted in the Final Office Action (p. 13) that:

Applicant has submitted that neither Knutson nor Seidman teach or suggest ‘deploying the application object contained in the application when ... absent from the list ...’ (Appl. Rmrk., pg. 11, middle). Knutson clearly teaches putting together files (Note: Knutson’s use of manifest file -- col. 4, lines 44-48-- implies that Jar files are listed by its descriptor) in a JAR package (see Fig. 6-7); and when it is time to redeploy, cached JAR files are scanned by mean of using descriptor matching. There would be no scanning of a package if there is only one entry; and a plurality of entries thus packaged and being scanned to identify some mismatch reads on a list and whether an application file is absent from a list. Applicant requires that ‘determining if an application file is absent from a list to avoid ... application object.’ The fact that Knutson scans a package of listed descriptor in order whether to redeploy the JAR files or to cache newly updated package amounts to the same result as recited but termed differently in the claim.

Applicants fail to see how scanning “a package of listed descriptor in order whether to redeploy the JAR files or to cache newly updated package” is the same as “deploying the application object... when the application file is absent from the list of all the files associated with previously deployed application objects” as required by Applicant’s independent claims 19, 38 and 45 (emphasis added).

In view of these arguments, Applicants respectfully request allowance of all of the claims.

(Applicant’s original response follows.)

III. Claims Rejected under 35 USC §103(a)

The Examiner rejected claims 1-54 under 35 USC §103(a) as being unpatentable in view of Knutson (U.S. Patent No. 6,557,100) and Seidman (U.S. Patent Publication No. 2003/0005166). Applicant respectfully traverses these rejections.

A *prima facie* obviousness rejection requires the Examiner to show that the prior art alone or in combination teaches or suggests all elements of the claimed invention. Applicant respectfully submits that the Examiner has failed to set forth a *prima facie* case of obviousness.

A. Independent Claims 1, 37 and 40

Independent claims 1, 37 (indirectly) and 40 recite in part (emphasis added):

deploying the undeployed application to a selected portion of the plurality of processing devices, such that the application is capable of being executed by the portion of the plurality of processing devices.

Knutson discloses a system for a caching Enterprise Java Beans (EJBs) such that when a EJB is deployed, a copy of it is cached on the server. Knutson, *Abstract*. Thereafter, if the EJB is redeployed, it is compared to the cached copy to determine if the associated deployment

descriptors or remote interfaces have changed. If there has been no change, then the cached copy is installed on the server without the need to regenerate and recompile the EJB code.

The Examiner admits that “Knutson does explicitly disclose that deploying of the undeployed application is deploying to selected portion of the plurality of processing devices.” See p. 3 of the Office Action. But the Examiner vaguely asserts that “Lan-based rules and session authentication” are equivalent to this feature of Applicant’s claim. However, Applicant notes that there is no mention of “LAN-based rules” or “session authentication” in Knutson. Furthermore, the relied upon text simply discusses a distributed computing environment, not the selection of processors to deploy applications on. The Examiner has failed to provide any evidence that would even suggest this feature of Applicant’s claims.

Furthermore, Seidman does not remedy the deficiency in Knutson. Seidman teaches deploying to a directory service, not a selected plurality of processing devices (Seidman, Abstract):

...the component manager including: a deployer that generates a client interface for each tracking component output port, and deploys the client interface in a directory service, wherein each entry is a tracking point object. In another embodiment, the deployer further generates a client interface for each tracking component input port, and deploys the client interface in a directory service, wherein each entry is a tracking point object.

Thus, Knutson and Seidman do not teach or suggest deploying an undeployed application to a selected portion of the plurality of processing devices. Accordingly, independent claims 1, 37 and 40 are not rendered obvious by the cited art for at least this reason. Claims 2-9 and 13-16 depend from claim 1. Claims 41-44 depend from claim 40. Accordingly, the dependent claims are patentable for at least the reason given in connection with the claims from which they depend.

B. Independent Claims 19, 38 and 45

Independent claims 19, 38 (indirectly) and 45 recite in part (emphasis added):

for each application file, deploying the application object contained in the application file when the application file is absent from the list of all the files associated with previously deployed application objects

This feature of Applicant's claims indicates that an application object is deployed if the application file for the object is not on a list of files associated with previously deployed application objects. In contrast, Knutson discloses that "if the EJB is redeployed, it is compared to the cached copy to determine if the deployment descriptors or remote interfaces have changed. If there has been no change, then the previously deployed EJB is installed on the server...." Knutson, col. 2, lines 40-48. Thus, Knutson teaches that a comparison of remote interfaces or deployment descriptors is required to avoid regenerating an EJB whereas Applicant's claims merely require determining if an application file is absent from a list to avoid redeploying an application object.

Seidman does not remedy the deficiency in Knutson. Seidman teaches that "DeploymentCoordinator 330 handles deployment of new TrackerBeans and tracking point networks. ... Whenever DeploymentCoordinator 330 detects a new TrackerBean deployment file, it reads each of the components in the deployment file, along with the component's associated deployment descriptors." Seidman, ¶¶ 27-28. Thus, Seidman implies that TrackerBeans are deployed regardless of whether they have been previously deployed.

Thus, Knutson and Seidman do not teach or suggest deploying the application object contained in the application file when the application file is absent from the list of all the files associated with previously deployed application objects. Accordingly, independent claims 19, 38 and 45 are not rendered obvious by the cited art for at least this reason. Claims 20-28 depend from claim 19. Claims 46-49 depend from claim 45. Accordingly, the dependent claims are patentable for at least the reason given in connection with the claims from which they depend.

IV. Conclusion

In view of the above, it is respectfully submitted that all of the Claims now pending in the subject patent application should be allowable, and reconsideration thereof is respectfully requested. The Examiner is respectfully requested to telephone the undersigned if he can assist in any way in expediting issuance of a patent.

The Commissioner is authorized to charge any underpayment or credit any overpayment to Deposit Account No. 06-1325 for any matter in connection with this response, including any fee for extension of time, which may be required.

Respectfully submitted,

Date: August 2, 2005

By:



Daniel J. Burns
Reg. No. 50,222

FLIESLER MEYER LLP
Four Embarcadero Center, Fourth Floor
San Francisco, California 94111-4156
Telephone: (415) 362-3800
Customer No. 23910